



The STI: Connecting Soar's Tools

Douglas Pearson

Scott Pultz

Soar Workshop 22

June 2002

The Problem

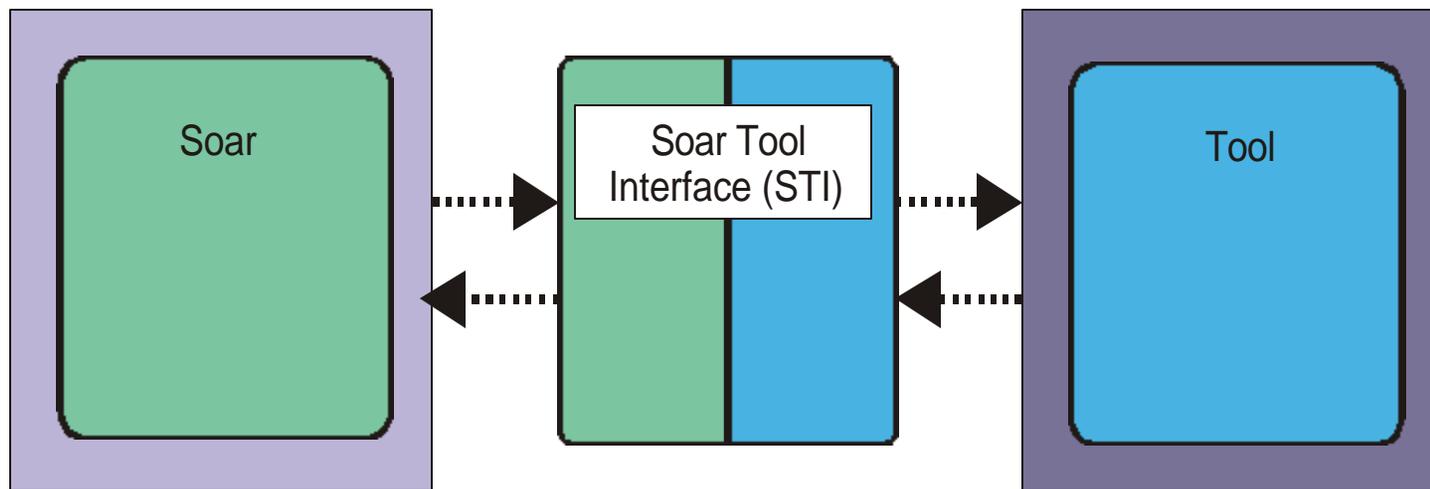
- Soar's tools are largely disconnected from Soar itself.
- For example, productions edited in Visual Soar are sent to Soar by cutting/pasting.

It's not a new problem

- There have been earlier integrated environments
 - SDE/Soar-mode : Built into Emacs.
 - Frank Ritter : Nice environment in Lisp.
- Environment tied directly to a version of Soar...but Soar changes rapidly.
- Tools proved difficult to maintain

The Goal

- To create a clean, general purpose interface between Soar and any tool.



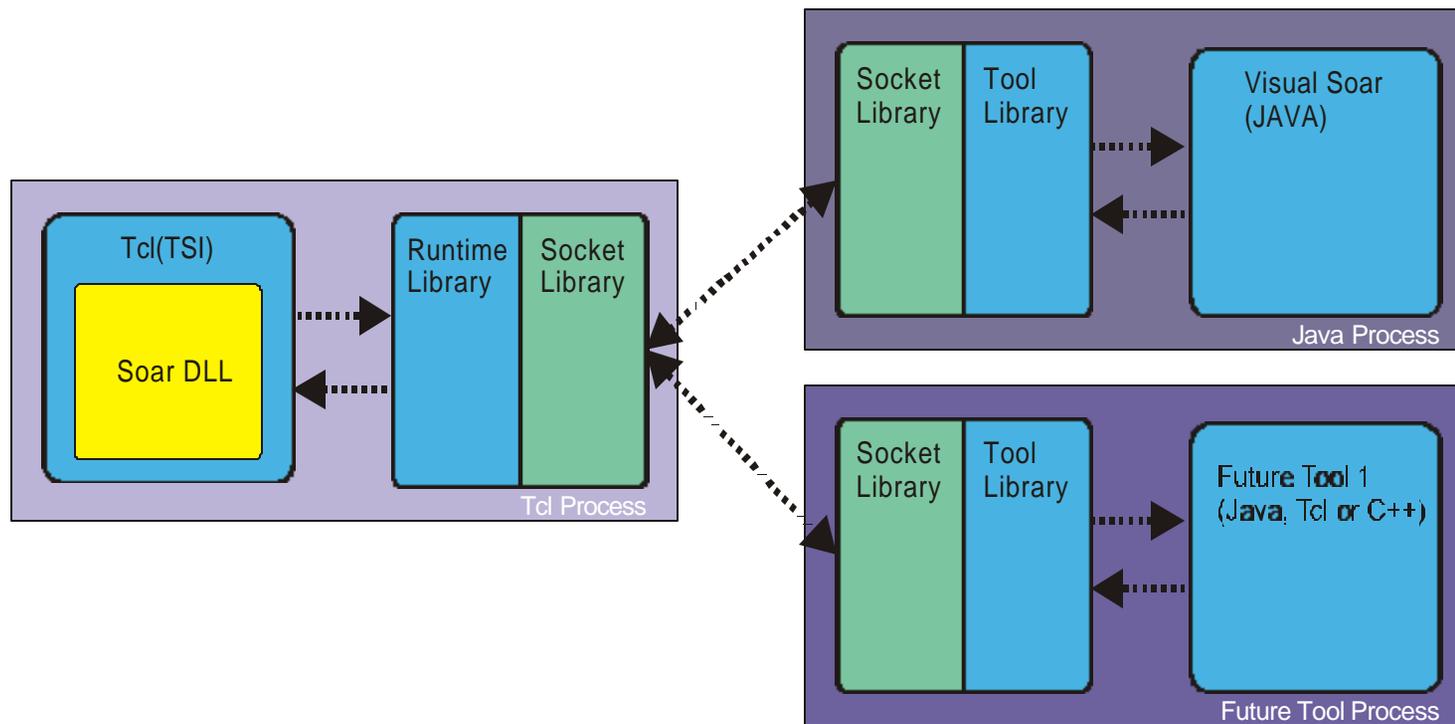
Design Principles

- Decoupled
- Clear Interface
- Language Flexibility
- Extensibility (Soar 8 -> Soar 9)
- Scalability
- Easy to Implement

Our Solution: The STI

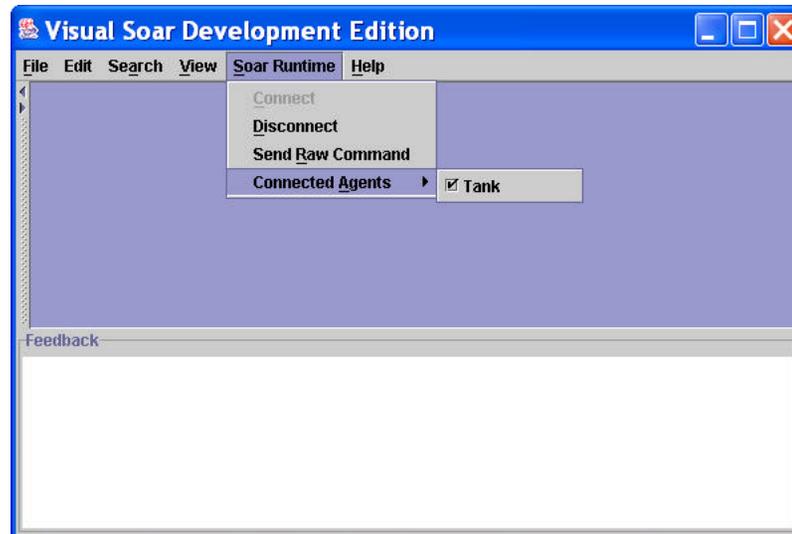
- Soar Interface Layer
 - Abstracts over the command set going to Soar and from Soar.
 - E.g. Send-production, edit-production.
- Socket Layer
 - Low level communication of data.

Soar Tool Interface



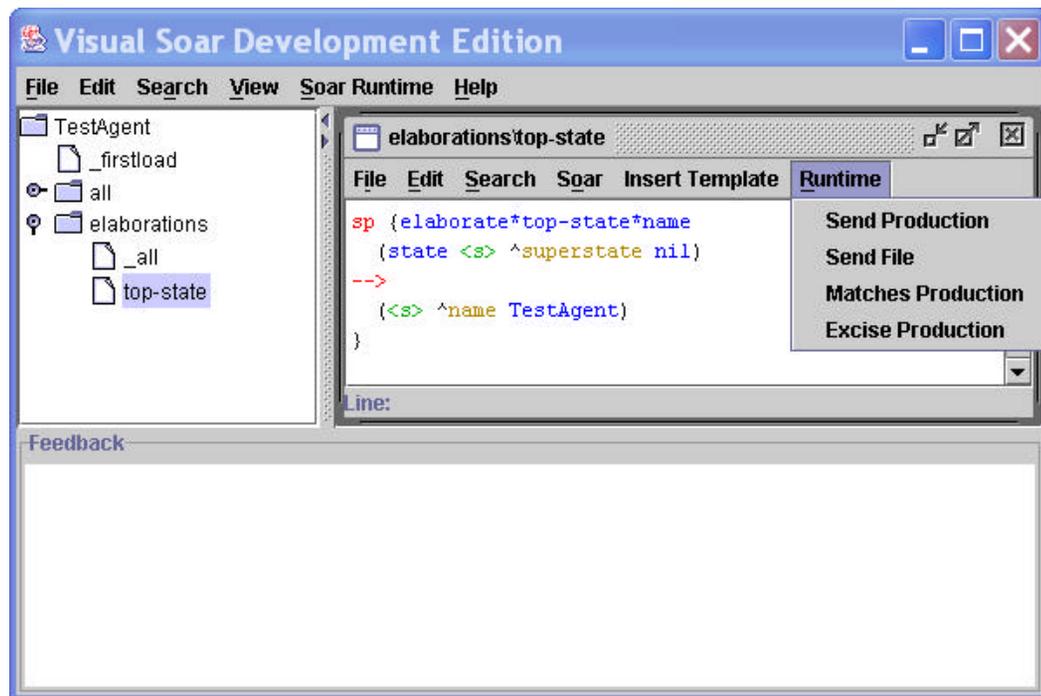
Visual Soar to Soar Example

- Launch Soar and Visual Soar
 - They automatically connect to each other.
 - Can select specific agents for Visual Soar to talk to



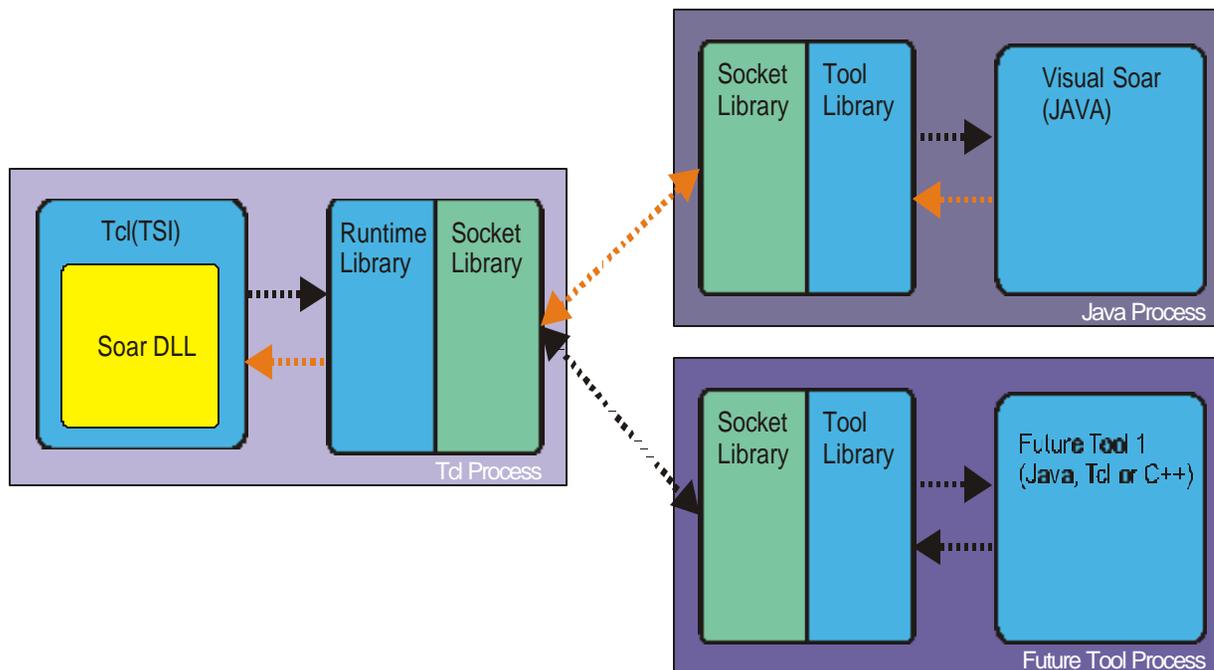
Visual Soar to Soar Example (2)

- Select commands in Visual Soar from menu



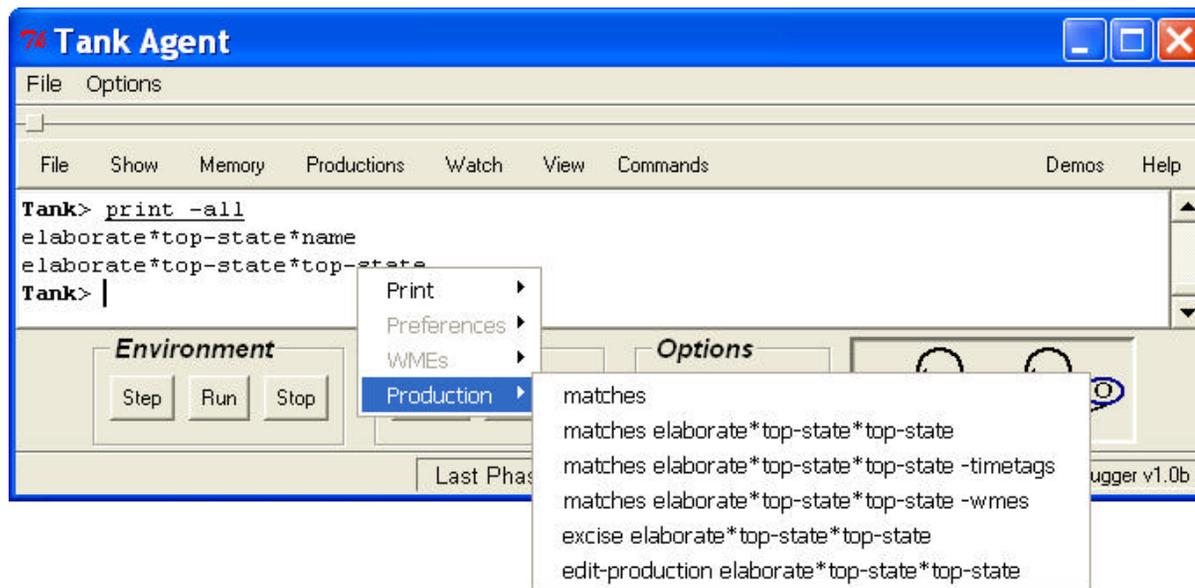
Visual Soar to Soar Example (3)

- Command is sent to Soar agent



Soar To Visual Soar

- Commands can be sent the other way too



Implementation

- Main interface layer and socket code written in C/C++, compiled into single shared library.
- Small additions to Visual Soar/TSI to connect to STI library.
- Tool-side logically very similar to Agent-side in code and interface.
- Lots of code re-use.

Implementation (2)

- Initial prototype implemented:
 - Send Production
 - Send File
 - Matches
 - Excise
 - Edit production (from Soar to tool)
 - Send raw command
 - Entire connection/messaging infrastructure.

Design Choices

- Interface library in C/C++
 - Almost all languages support calls to C
 - Separate library decouples code from specific Soar version
- Socket layer
 - Solves cross process communication problem
 - Simple and widely available
 - Could be replaced later if necessary

Meeting Design Goals

- **Decoupled** – Tool doesn't talk directly to Soar
- **Clear Interface** – Explicit interface layer
- **Language Flexibility** – Universal calls to C
- **Extensibility** – Either side can change more easily
- **Scalability** – Commands all take constant time
- **Easy to Implement** – 1 month

Next steps

- Extending the command set
- Improving feedback as commands are executed
- Extending commands to support parameters
- Saving user choices between sessions
- Interface to “G-Ski” instead of TSI?
- Lots more...

More Information

- <http://www.threepenny.net/~soartech/download.html>
 - Latest version of STI.
- Full spec available.
- Email: doug@threepenny.net
- Email: scott@threepenny.net